CASIS: A System for Concept-Aware Social Image Search*

Ba Quan Truong
bqtruong@ntu.edu.sg

Aixin Sun
axsun@ntu.edu.sg

Sourav S. Bhowmick
assourav@ntu.edu.sg

School of Computer Engineering, Nanyang Technological University, Singapore 639798

ABSTRACT

Tag-based social image search enables users to formulate queries using keywords. However, as queries are usually very short and users have very different interpretations of a particular tag in annotating and searching images, the returned images to a tag query usually contain a collection of images related to multiple *concepts*. We demonstrate CASIS, a system for concept-aware social image search. CASIS detects *tag concepts* based on the collective knowledge embedded in social tagging from the initial results to a query. A tag concept is a set of tags highly associated with each other and collectively conveys a semantic meaning. Images to a query are then organized by tag concepts. CASIS provides intuitive and interactive browsing of search results through a *tag concept graph*, which visualizes the tags defining each tag concept and their relationships within and across concepts. Supporting multiple retrieval methods and multiple concept detection algorithms, CASIS offers superior social image search experiences by choosing the most suitable retrieval methods and concept-aware image organizations.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information Filtering*

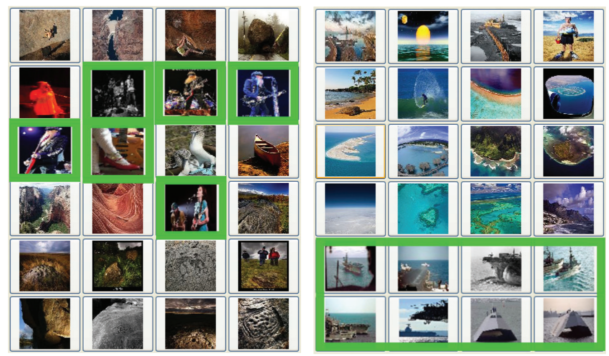
Keywords

Flickr, TagIR, Tag Concept, Tag Relation, Image Search

1. INTRODUCTION

The availability of social tags on various image sharing platforms (*e.g.,* Flickr) raises the opportunity of building effective tag-based social image retrieval systems. In contrast to *content-based* image retrieval paradigm, which searches for visually similar images of a given query image, *tag-based* image retrieval (TAGIR) enables users to formulate semantic queries using textual keywords through an interface similar to Web search engines. Interestingly, similar to general Web search queries, the queries for tag-based image search are usually very short, consisting of 2.2 tags on average for each search [10].

A short query, particularly a single-tag query, usually leads to a large number of potentially relevant images. More importantly, these images often represent multiple *concepts* because: (i) a single-tag has limited expressiveness in precisely defining the information need, and (ii) social tags are noisy in nature as different users might have very different understandings of a particular tag in annotating and searching images. Figures 1(a) and 1(b) depict the search results of two sample single-tag queries, Q_r(`rock`) and Q_s(`sea`), respectively[1]. Observe that the results of Q_r capture two different concepts. Specifically, the highlighted images (in green border) are about rock music while the other images are related to stone rock or mountain. Similar phenomenon is observed for Q_s. Although most images returned for Q_s are good matches to the query, the highlighted ones are mainly related to ship and the remaining ones deal with seascape (beaches, sky, clouds, etc.). This multiple concepts phenomenon might be attributed to the ambiguity of the query itself (*e.g.,* `rock`) or the different aspects of the query (*e.g.,* `sea`), among other reasons. Importantly, a user is usually interested in *only one* of the concepts and not all of them. *Consequently, displaying images representing multiple concepts in a single ranked list may adversely affect image search experiences.*



(a) Query Q_r(`rock`) (b) Query Q_s(`sea`)

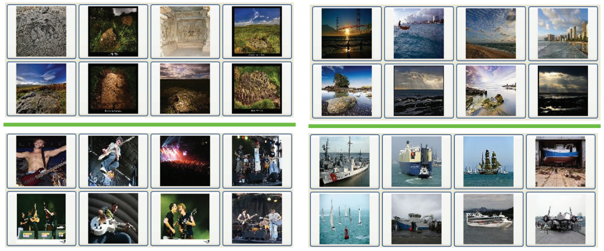
Figure 1: Results for single-tag queries: Q_r(`rock`), Q_s(`sea`).

In this demonstration, we present a social image search system called CASIS (**c**oncept-**a**ware **s**ocial **i**mage **s**earch), which exploits the notion of *tag concept* (or simply *concept*) to address the aforementioned problem. The intuition is that a specific meaning or aspect of a short (particularly single-tag) query can be well described by a group of highly *related* tags. Each such group of tags is referred to as a *tag concept*. Accordingly, images matching the query can be organized into groups, each of which matches one tag concept. For instance, Figure 2 depicts the reorganization of the search results of Q_r(`rock`) and Q_s(`sea`) in CASIS according to their corresponding tag concepts. Note that, CASIS supports queries consisting

*This work was supported by Singapore MOE AcRF Tier-1 Grant RG13/10.
CASIS webpage: `http://www.ntu.edu.sg/home/axsun/casis.html`

[1]For clarity, we recommend viewing all figures presented in this paper directly from the color PDF, or from a color print copy.

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.



(a) Rock stone and Rock music (b) Seascape and Ship

Figure 2: Results matching concepts for Q(`rock`) and Q(`sea`).

of single or multiple tags. In this paper, we mainly use single-tag queries for illustration purpose.

Intuitively, for a given query, CASIS retrieves a set of images best matching the query mainly based on the relevance scores between the query tag(s) and the images. This list of images serves as the initial search results to be presented to the user. From the initial results, the co-occurring relationships among the frequent tags are identified to form a *Tag Relation Graph* (TRG). A node in a TRG is a tag and the weight of an edge between two tags represents the *strength* of the co-occurrence of the two tags. Figure 5(a) gives an example TRG for Q_s(`sea`). A graph-cut algorithm is then applied to (softly) cut the TRG to form a *Tag Concept Graph* (TCG). A TCG is a graph with meta-nodes, each representing a group of tags. Tags in each group, linked by co-occurrence relationships, jointly represent a tag concept. The links across meta-nodes indicate the relationships between tag concepts. Note that two tag concepts associated with a query may be unrelated (*e.g.,* rock music and rock stone) or related (*e.g.,* ship and seascape) to each other. The concepts may also *overlap* if they share some identical tags. A tag concept may even subsume another by containing all the latter's tags as semantically concepts can be defined at different levels of abstraction. For visualization, TRG and TCG are superimposed to illustrate the relationships between tags and tag concepts (See Figure 5).

The benefit of CASIS is two-fold. First, the search results in CASIS are better organized into groups where each group of images is relevant to a more cohesive tag concept. Note that an end user may not always be aware of the existence of such concepts due to the collaborative nature of social tagging. Different users may use the same tag for different purposes and in very different manners. Hence, the tag concepts provide a superior mechanism to browse the search results. Second, by automatically identifying the tag concepts, the original tag query can be enriched by adding the concept-specific tags. As the information need is now much more clearly defined, accurate image search results are expected for each concept-enriched tag query. Note that in [10] we have empirically demonstrated that the search accuracy of a single-tag query is largely affected by the choice of *tag relatedness* measure (*i.e.,* the effectiveness of a tag in describing its annotated image) and the matching model between the query tag and an image's tag (see Section 2 for more details). However, for a multi-tag query, the presence of all query tags in an image largely guarantees a very good match. Thus, CASIS facilitates refinement of the original query by adding concept-specific tags as well as retrieval of concept-relevant images within the initial search results of the query.

Clustering search results and presenting the results in groups have been studied in traditional Web search setting (see [2] for a survey). However, there are at least two key differences between Web clustering engines and concept-aware TAGIR. Firstly, there exists a fundamental difference in the way users perceive text and image data. In Web search engines, a user needs to read the title/snippet of a hit to judge its relevance, one by one. In contrast, a glance at image thumbnails could easily tell a user whether a set of images are relevant. Secondly, Web search results and social images are represented in very different feature spaces which calls for different clustering techniques. These two key differences pose new challenges toward realizing concept-aware social image search. Particularly, it is important for CASIS to provide an intuitive user interface to facilitate quick judgment of relevance as well as the flexibility in browsing multiple tag concepts.

2. RELATED WORK

Image Search Results Clustering. Most approaches for clustering image search results exploit low-level visual features [3]. However, these approaches suffer from two problems: (a) semantic gap between the low-level features and high-level semantics and (b) low efficiency due to curse of dimensionality. IGroup [12] took a step towards addressing these limitations by exploiting textual features such as image captions, snippets, surrounding texts. The clustering is then accomplished by combining both visual features and textual features. In the context of social tagged images, *shared nearest neighbors* algorithm (SNN) was applied to cluster images in a collection using both tag features and visual features [5]. In CASIS, we aim to detect the concepts associated with a tag query and group the images according to the detected concepts. To be detailed in the next section, CASIS is flexible enough to easily accommodate different clustering algorithms (SNN could be one of them).

Tag-based Image Browsing. Using tags in searching and browsing social images has been explored from multiple dimensions. The systems presented in [4] and [1] explore temporal and spatial dimensions, respectively. Other systems utilize multi-faceted browsing to produce coherent image groups. The facets can be manually specified [11] or automatically detected using WordNet [7] or Wikipedia [6]. MediaFaces [11] extracts location, celebrity, movie entities from pre-determined sources such as Yahoo!Travel, GeoPlanet, Wikipedia, etc. A query is then mapped to a stored entity whose related facets are then used to produce groups of semantically related images. Olive [7] extracts concepts from WordNet nouns for faceted browsing. WordNet, however offers relatively low coverage for the fast-changing keywords such Flickr tags. To address this problem, Wikipedia is considered in [6] for improving the coverage which is then used in TagExplorer [8], a faceted browsing system for Flickr photos. Different from faceted browsing where the facets are often pre-determined, CASIS detects tag concepts automatically in real-time from the tagged images.

Tag-based Image Searching. Concept-aware social image search relies on an effective TAGIR ranking method to retrieve the images that best match the query before these images can be grouped into different concepts. In CASIS we adopt the TAGIR framework consisting of five orthogonal dimensions for flexibly defining a specific TAGIR ranking method [10]. The five dimensions include *tag relatedness* for measuring the degree of effectiveness of a tag describing the tagged image, *tag discrimination* for quantifying the degree of discrimination of a tag with respect to the image collection, *tag length normalization* analogous to document length normalization in Web search, *tag-query matching model* for computing matching score between an image tag and a query tag, and *query model* for *rewriting* tag queries. A systematic evaluation of hundreds of TAGIR methods is reported in [10]. In [9], we show that tag concepts can be used to improve image tag recommendation.

3. SYSTEM OVERVIEW

In this section, we give an overview of the CASIS system. Figure 3 depicts the system architecture. The *tagged image retriever*

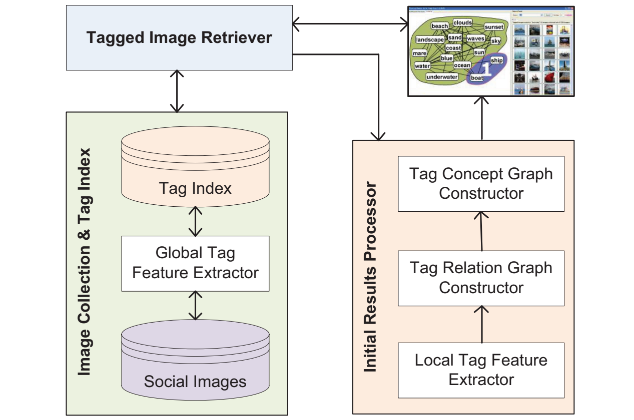


Figure 3: System architecture of CASIS.

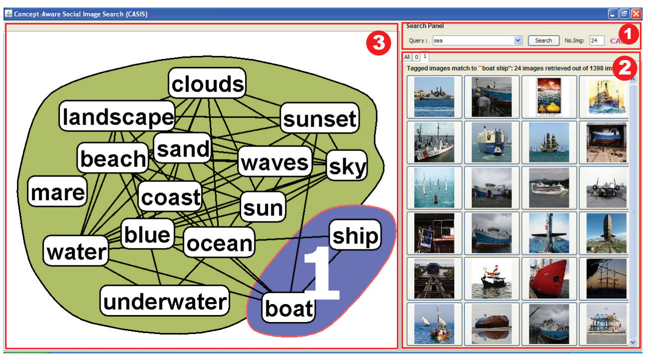


Figure 4: Screenshot of the GUI of CASIS for query Q_s(sea).

module retrieves a relatively large number of images for further processing by the *initial results processor*. The latter delivers the final results to the user through the user interface. We elaborate on various modules and components in turn.

3.1 User Interface

Recall that the user interface design is crucial in addressing the challenges in concept-aware TAGIR. Figure 4 depicts the the main GUI of CASIS using the query Q_s(`sea`). It consists of three panels. User issues a tag query by keying keyword(s) and the desired number of h image hits in the search panel (Panel 1). Clicking on the CASIS label (purple color) will invoke the *configuration dialog box* to set various parameters. Once the query is processed, the top-h image hits are displayed in the first tab (labeled `"All"`) in result panel (Panel 2). When we mouse over each image, its associated tags are displayed sorted by descending order of tag relatedness. Clicking on tab *id* on the top-left corner of the result panel displays the top-h image hits for the concept *id*. For example, in Figure 4, the image hits for the tag concept ship, boat are displayed in Panel 2. The corresponding tag concept *id* is highlighted in the tag concept graph visualized in Panel 3, showing the relationship between tags within and across the concepts. Mouse hovering on the other parts of the TCG brings up other concept *id*s. Double-clicking a tag in the graph refines the query by adding the tag into the query box in Panel 1. The TCG visualization is implemented using `Prefuse` package[2] allowing color-coded tag concepts with overlapping and containment relationships. Mouse hovering a large concept may reveal sub-concepts contained in it. Tag concepts are visualized using different background colors.

3.2 Tagged Image Retriever Module

Given a query Q containing one or more query keywords, this module retrieves top-N images that best match Q where N is the predefined size[3] of initial search results (default setting N=5000). Most TAGIR algorithms can be adopted by this module, and in our implementation, we adopt the framework in [10] for supporting various settings in TAGIR (see Section 2). This module relies on two databases, the tagged images and the tag index where the tag features are extracted by *Global Tag Feature Extractor*. The indexes are implemented using `MySQL` and `Lucene`, respectively. The images in the initial search results are then indexed in an in-memory Lucene index for further processing. This in-memory index will be used to search for concept-specific images.

Global Tag Feature Extractor. This component extracts query-independent tag features (*e.g.,* tag frequency, tag relatedness, tag associations, etc.) from the collection of social images. We briefly discuss a subset of features used in CASIS. *Tag frequency* of a tag t is the number of images annotated with t. *Tag co-frequency* between two tags t_1 and t_2 is the number of images annotated by both t_1 and t_2. These two features are used to compute tag associations (or co-occurrences) using different measures (*e.g.,* Jaccard coefficient, Pointwise Mutual Information, Pointwise KL divergence). To support the framework in [10], the tag relatedness between an image and any of its annotated tags is computed and stored. The default *tag relatedness* is by neighbor voting. A tag t receives a high relatedness score towards image d if many of the nearest neighbors of d by visual similarity are also annotated by t.

3.3 Initial Results Processor Module

This module is responsible for constructing query-dependent tag relation graph and tag concept graph. We detail the three major components in this module.

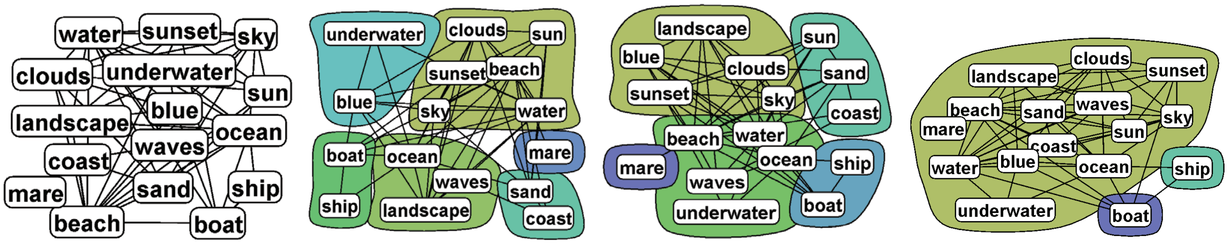
Local Tag Feature Extractor. This component first extracts the tags that are most related to the query tags from the initial results. Hence the tags and their features extracted are query-dependent. The features for the tags include tag frequency and tag co-frequency for constructing tag relation graph. Given that there can potentially be a large number of distinct tags associated with all the images in the initial results, the key issue here is to determine which tags to consider for subsequent processing.

Let T_d be the set of tags associated with an image d. For each image, we extract its top-r related tags ordered by tag relatedness, where $r = max(\tau, \lceil \rho * |T_d| \rceil)$. Both τ and ρ are configurable in CASIS ($\rho = 0.1$ and $\tau = 4$ by default). In other words, for each image, we want to consider a reasonably small set of tags that best visually describe the image. The main reason is that tags are noisy in nature and as a result many of them do not effectively describe the images. On the other hand, some images may not be well tagged and $|T_d|$ can be very small; τ is introduced to avoid a very small r. The extracted tags are then considered as *candidate tags* for constructing the tag relation graph, and subsequently tag concept extraction.

Tag Relation Graph Constructor. This component constructs the *tag relation graph* (TRG) from the initial results. The nodes in a TRG are the most frequent tags among the candidate tags identified by the local tag feature extractor with respect to the tag frequency in the entire dataset. Let D_Q be the initial results for query Q, and $\mathcal{D}$ be the entire dataset. The *normalized relative frequency* of a tag t

[2] `http://prefuse.org/`

[3] If a query has fewer than N matching images, then all matching images are retrieved.



(a) Tag relation graph (b) K-Nearest Neighbor (c) Affinity Propagation (d) Markov Clustering

Figure 5: Tag relation graph for Q_s(sea), and the tag concepts detected by different clustering algorithms.

is $\frac{df(t,D_Q)}{|D_Q|} - \frac{df(t,\mathcal{D})}{|\mathcal{D}|}$, where $df(t,S)$ denotes the number of images in set S annotated by t. The number of nodes in a TRG is $n \times \log(|D_Q|)$ where n is configurable and set to 1.5 by default.

An edge between two tags in a TRG can be computed using any first-order co-occurrence measure. CASIS also supports second-order co-occurrence. Specifically, for each tag in the TRG node set, we get its top-ℓ most associated tags within the candidate tags by a selectable first-order co-occurrence measure ($\ell = 20$ by default for efficiency). Together with the co-occurrence scores, each tag is represented by a vector with ℓ non-zero entries. The second-order co-occurrence between two tags is computed by cosine similarity between two such vectors.

With second-order co-occurrence, edge weights between any pair of nodes in a TRG are likely to be non-zero, making the TRG almost a complete graph. Thus, the edges are further filtered using k-nearest neighbor and ε-neighborhood models. The k-nearest neighbor model keeps top-k highest weight edges for each node. The ε-neighborhood model prunes all edges whose weights are below a threshold ε. In CASIS, we set $k = 5$ and ε equals to the median of all edge weights.

Tag Concept Graph Constructor. Finally, this component detects tag concepts from the TRG using an existing graph-cut or clustering algorithm. Notably, this component is designed to seamlessly incorporate different clustering algorithms (both hard and soft clustering) with different configurations, allowing users to explore different possible concepts. CASIS currently supports five clustering algorithms, namely, *Border Flow* (BF), *Chinese Whisper* (CW), *K-Nearest Neighbor* (KNN), *Affinity Propagation* (AP), and *Markov Clustering* (MCL) (mainly based on the `Cluster Visualization Kit`[4]). As it is unrealistic to predict the number of concepts in a search, we choose those clustering algorithms which do not require the number of pre-defined clusters. Among them, BF, CW, and MCL support soft-clustering enabling overlapping and subsumed tag concepts. Figures 5(b), 5(c), and 5(d), show the concepts detected for Q_s(`sea`) by KNN, MCL, and AP, respectively. The concepts detected by BF are shown in Figure 4. Note that, the query tag (*i.e.,* `sea`) is not shown in TRG or TCG as it is related to all tags and all concepts.

For each detected concept, its tags are used to refine the query to get the concept-specific results from the initial results, and then display them under a numbered tab (*id*) in the result panel. That is, CASIS presents both the tags (and their relationships) defining each concept and the corresponding image results for the concept.

4. DEMONSTRATION OVERVIEW

Our demonstration will be loaded with NUS-WIDE dataset. The original tags (without cleaning) of images are used in this demonstration. All global features (*e.g.,* tag frequency, tag association, tag relatedness) are pre-computed by Global Tag Feature Extractor. Using this dataset, we aim to showcase the functionality and effectiveness of CASIS in identifying multiple tag concepts and presenting concept-aware TAGIR results. Specifically, we shall showcase the followings.

Tag Relation Graph and Concept Graph. Through our GUI, we shall demonstrate how CASIS identifies closely related tags as concepts from the initial query results, and visualizes tag relationships within and across tag concepts. We shall also showcase the effect of various tag association measures and clustering algorithms for concept detection problem.

Concept-Aware Image Search and Results Browsing. Using CASIS, we shall demonstrate concept-aware image search in action and real-time visualization of results categorized by concepts. Recall that tags in concepts are also good candidate tags for query refinement. Hence, end users will also be able to interactively experience superior *topic-specific* social image search when they select some of the tags (by clicking nodes in TCG) to refine their queries.

[4] `http://borderflow.sourceforge.net/`

5. REFERENCES

[1] S. Ahern, M. Naaman, R. Nair, and J. H.-I. Yang. World explorer: visualizing aggregate data from unstructured text in geo-referenced collections. In *JCDL*, pages 1–10, 2007.

[2] C. Carpineto, S. Osiński, G. Romano, and D. Weiss. A survey of web clustering engines. *ACM Comput. Surv.*, 41:17:1–17:38, July 2009.

[3] Y. Chen, J. Z. Wang, and R. Krovetz. Content-based image retrieval by clustering. In *MIR*, pages 193–200, 2003.

[4] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, and A. Tomkins. Visualizing tags over time. In *WWW*, pages 193–202, 2006.

[5] P.-A. Moëllic, J.-E. Haugeard, and G. Pitel. Image clustering based on a shared nearest neighbors approach for tagged collections. In *CIVR*, pages 269–278, 2008.

[6] S. E. Overell, B. Sigurbjörnsson, and R. van Zwol. Classifying tags using open content resources. In *WSDM*, pages 64–73, 2009.

[7] A. Popescu and P.-A. Moëllic. Olive: a conceptual web image search engine. In *ACM Multimedia*, pages 147–148, 2007.

[8] B. Sigurbjörnsson and R. van Zwol. TagExplorer: Faceted browsing of flickr photos. Technical Report YL-2010-005, Yahoo! Research.

[9] A. Sun, S. S. Bhowmick, and J.-A. Chong. Social image tag recommendation by concept matching. In *ACM Multimedia*, pages 1181–1184, 2011.

[10] A. Sun, S. S. Bhowmick, K. T. Nam Nguyen, and G. Bai. Tag-based social image retrieval: An empirical evaluation. *JASIST*, 62(12):2364–2381, 2011.

[11] R. van Zwol, B. Sigurbjörnsson, R. Adapala, L. G. Pueyo, A. Katiyar, K. Kurapati, M. Muralidharan, S. Muthu, V. Murdock, P. Ng, A. Ramani, A. Sahai, S. T. Sathish, H. Vasudev, and U. Vuyyuru. Faceted exploration of image search results. In *WWW*, pages 961–970, 2010.

[12] S. Wang, F. Jing, J. He, Q. Du, and L. Zhang. Igroup: presenting web image search results in semantic clusters. In *CHI*, pages 587–596, 2007.